

Table of Contents

Journal.....	2
3 Tasks.....	3
3.1 Build OpenSudoku.....	3
3.2 Prune OpenSudoku.....	7
Screenshots Before Pruning.....	9
Screenshots After Pruning.....	10
Firebase Getting Started.....	11
3.3 Firebase Storage.....	12
3.4 Firebase Auth.....	14
3.5 GitHub.....	17

CEG 3900 Mobile and Cloud Computing P2

Submitted by Paul Fuchs (U00747698) on 10 February 2017

<https://github.com/helmmhammerhand/ceg-3900-spring-2017>

Journal

Firebase Setup (1/2 hour)

Created a Firebase account and created storage and auth projects and retrieved google-services.json files

Firebase Auth and Storage (2 hours)

The projects from github compiled without too much modification but I did have trouble running it. First, it did not work on my emulator because of its Google Play version. When running on my phone, it gave an authentication error until I configured the authentication options in Firebase as detailed in a Pilot discussion post.

Build Open Sudoku (1 hour)

This one did not take that long as I started it for P1. I ended up removing the menu bar because it kept causing runtime errors when inflating.

Prune Open Sudoku (2 ½ hours)

Mostly consisted of deleting files and a few minor code changes to allow the app to run without the extra files.

Github (1+ hour)

I have not used github before and this is taking me a decent amount of time to figure out how to use.

Writeup (3 hours)

Consisted of organizing deliverables, adding screenshots to this file and writing about the tasks

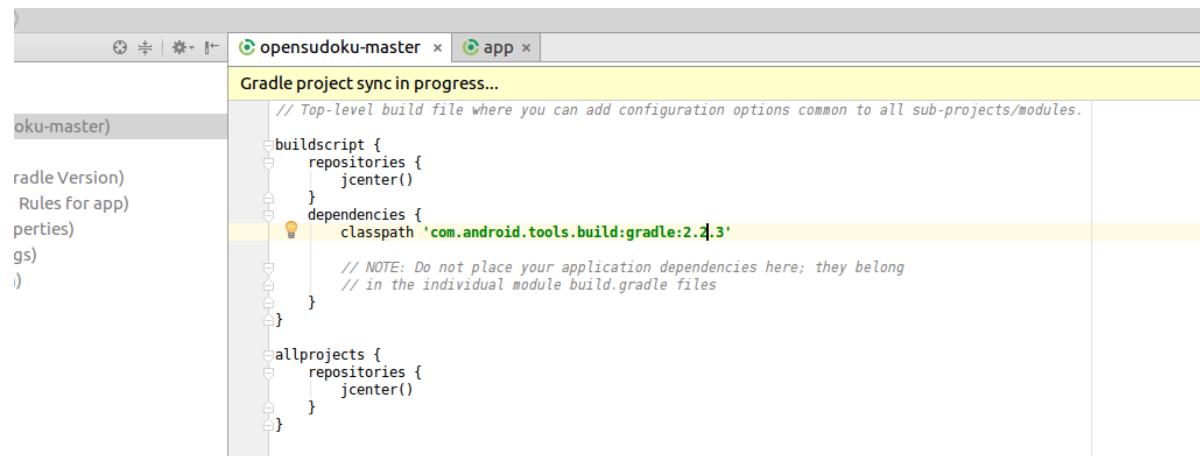
3 Tasks

3.1 Build OpenSudoku

The following modifications were made to the open sudoku code in order to build it.

1. Change gradle build versions

In opensudoku-master/build.gradle, the gradle classpath was set to 2.2.3 as shown below:



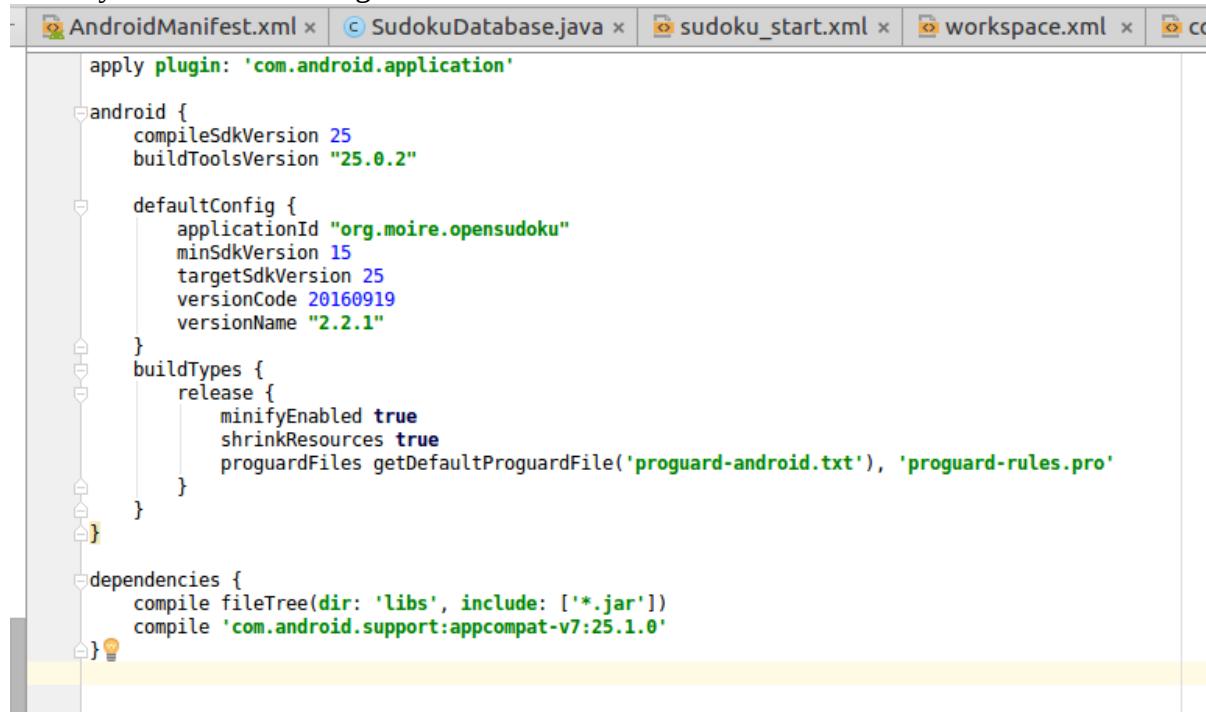
```
// Top-level build file where you can add configuration options common to all sub-projects/modules.

buildscript {
    repositories {
        jcenter()
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:2.2.3'

        // NOTE: Do not place your application dependencies here; they belong
        // in the individual module build.gradle files
    }
}

allprojects {
    repositories {
        jcenter()
    }
}
```

In app/build.gradle, the compileSdkVersion, buildToolsVersion, targetSdkVersion, and appcompat library version were changed to the values shown below:



```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 25
    buildToolsVersion "25.0.2"

    defaultConfig {
        applicationId "org.moire.opensudoku"
        minSdkVersion 15
        targetSdkVersion 25
        versionCode 20160919
        versionName "2.2.1"
    }
    buildTypes {
        release {
            minifyEnabled true
            shrinkResources true
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
}

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    compile 'com.android.support:appcompat-v7:25.1.0'
}
```

After making these changes, the project built but produced the following run-time error when launched

```
AVD: Nexus_5_API_25 [app]
----- beginning of crash
FATAL EXCEPTION: main
Process: org.moire.opensudoku, PID: 2817
java.lang.RuntimeException: Unable to start activity ComponentInfo{org.moire.opensudoku/org.moire.opensudoku.gui.FolderListActivity}:
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2665)
    at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2726)
    at android.app.ActivityThread.-wrap12(ActivityThread.java)
    at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1477)
    at android.os.Handler.dispatchMessage(Handler.java:102)
    at android.os.Looper.loop(Looper.java:154)
    at android.app.ActivityThread.main(ActivityThread.java:6119) <1 internal calls>
    at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:886)
    at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:776)
Caused by: android.view.InflateException: Binary XML file line #31: Binary XML file line #31: Error inflating class com.android.inte
Caused by: android.view.InflateException: Binary XML file line #31: Error inflating class com.android.internal.widget.ActionBarConta
Caused by: java.lang.reflect.InvocationTargetException
    at java.lang.reflect.Constructor.newInstance0(Native Method) <1 internal calls>
    at android.view.LayoutInflater.createView(LayoutInflater.java:645)
    at android.view.LayoutInflater.createViewFromTag(LayoutInflater.java:787)
    at android.view.LayoutInflater.createViewFromTag(LayoutInflater.java:727)
    at android.view.LayoutInflater.rInflate(LayoutInflater.java:858)
    at android.view.LayoutInflater.rInflateChildren(LayoutInflater.java:821)
    at android.view.LayoutInflater.inflate(LayoutInflater.java:518)
    at android.view.LayoutInflater.inflate(LayoutInflater.java:426)
    at android.view.LayoutInflater.inflate(LayoutInflater.java:377)
    at com.android.internal.policy.DecorView.onResourcesLoaded(DecorView.java:1813)
    at com.android.internal.policy.PhoneWindow.generateLayout(PhoneWindow.java:2554)
    at com.android.internal.policy.PhoneWindow.installDecor(PhoneWindow.java:2627)
    at com.android.internal.policy.PhoneWindow.setContentView(PhoneWindow.java:402)
    at android.app.Activity.setContentView(Activity.java:2414)
    at org.moire.opensudoku.gui.FolderListActivity.onCreate(FolderListActivity.java:89)
    at android.app.Activity.performCreate(Activity.java:6679)
    at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1118)
    at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2618)
    at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2726)
    at android.app.ActivityThread.-wrap12(ActivityThread.java)
    at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1477)
    at android.os.Handler.dispatchMessage(Handler.java:102)
    at android.os.Looper.loop(Looper.java:154)
    at android.app.ActivityThread.main(ActivityThread.java:6119) <1 internal calls>
    at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:886)
```

TODO Android Monitor Terminal Messages

Run re-installed and restarted the app // (Dont show again) (moments ago)

A google search made it seem like this had something to do with the theme. After a lot of trial and error, I managed to get the apk to compile after making changes to the app theme.

2. Change from ThemeOverlay.AppCompat.Dark.ActionBar to Theme.AppCompat.Light.ActionBar in themes.xml



This resulted in an error because open-sudoku already contained a theme named Theme and the new theme value began with Theme.

3. Changed theme names from Theme to AppTheme because Theme is also a built in package which I was unable to use due to overloading. In the below screenshot, every occurrence of 'AppTheme' used to be just 'Theme'.



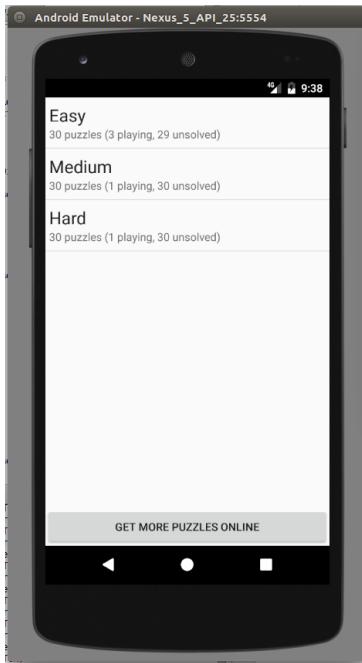
These changes had the side effect of making the menu not show up but stopped the app from crashing.

4. Removed a line of code from IMPopupDialog that referenced the nonexistent menu and was causing a NullPointerException.

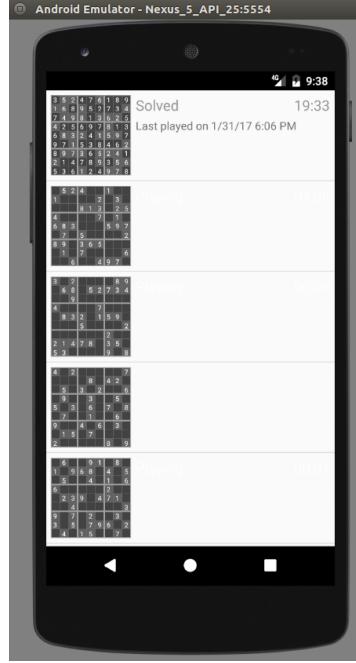
Screenshots of my build are shown on the next page.

From the version I built (like the Google Play one but missing menu bar and using light theme)

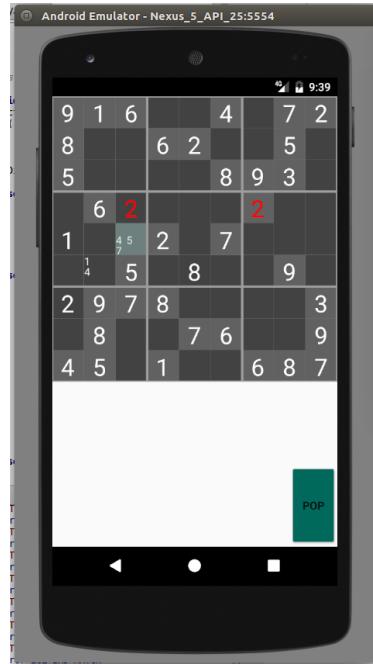
Starting Screen



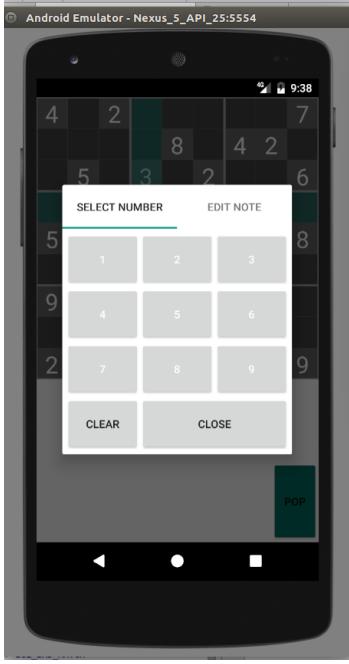
Selecting a Puzzle



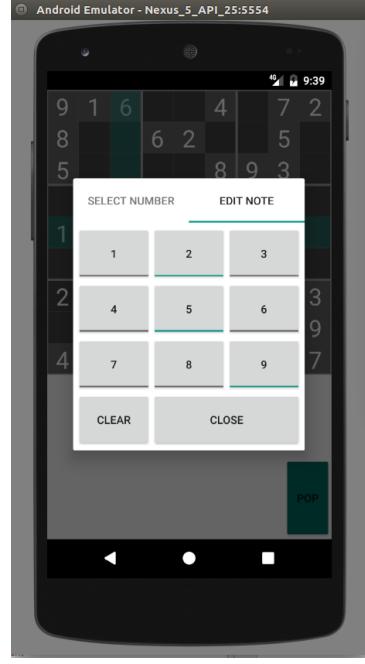
Playing the Game



Entering Numbers



Entering Notes (possible numbers)



3.2 Prune OpenSudoku

Summary of Changes

Files that Were Kept

SudokuPlayActivity: Allows the user to play a sudoku game. I removed the menu bar, made onCreate() always load puzzle 1, and disabled the saving of puzzles (collaborative puzzles will come from the server).

SudokuBoardView: responsible for displaying the board on screen.

gui/inputmethod/: contains different Java guis that let the user enter numbers on the Sudoku board

db/: Contains classes that allow puzzles to be stored, created, and saved. I am not using most of its functionality but left it because it might be handy when building the server.

HintsQueue: Used to display various messages to the user

Pruned Activities

FolderListActivity: user chooses group of puzzles Easy, Medium, or Hard. Also imports and exports puzzles and adds folders (I think). Starts a SudokuListActivity when user selects a folder.

SudokuListActivity: displays a scrolling list of puzzles which a user can select. Also displays info such as when the puzzle was solved. Also seems to allow user to reset or delete a puzzle (again via menu bar). When a puzzle is selected, it starts SudokuPlayActivity with the ID of the selected puzzle. I think this may be a good place to integrate firebase login, ect. Even if I end up deleting the ability to choose a puzzle.

GameSettingsActivity: not useful because I removed the menu to access it

SudokuImportActivity: imports puzzles from phone or internet

ImportSukokuActivity: old version of SudokuImportActivity kept for backwards compatibility

SudokuExportActivity: exports sudoku puzzles to phone

SudokuEditActivity: allows user to make custom boards

SudokuListFilter: filters puzzles by not started, playing, and solved. Used by SudokuListActivity

FileImportActivity: allows user to add a puzzle from an external file

gui/importing/: Reads puzzles from xml files

gui/exporting/: Writes puzzles to files

All Menu bar actions. These wouldn't run properly when I compiled so they will be deleted

All activities had their corresponding xml files deleted and removed from the Android manifest.

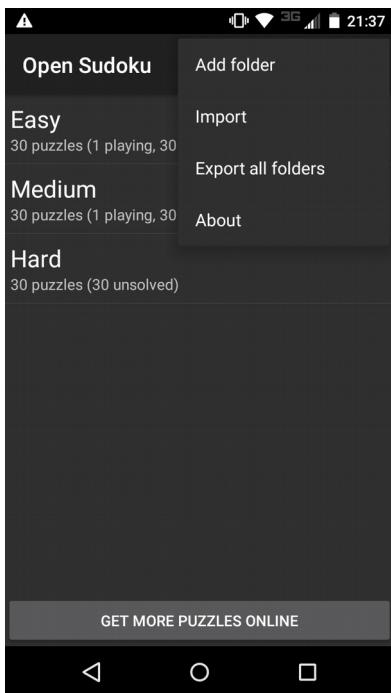
I added a class SudokuStartActivity to serve as the main activity of the apk. SudokuStartActivity contains credits and a button which launches SudokuPlayActivity. For now, I have made SudokuPlauActivity always load puzzle 1 upon starting. When changing to collaborative sudoku, this will be replaced with a request that gets the current sudoku board for the server.

Screenshots Before Pruning

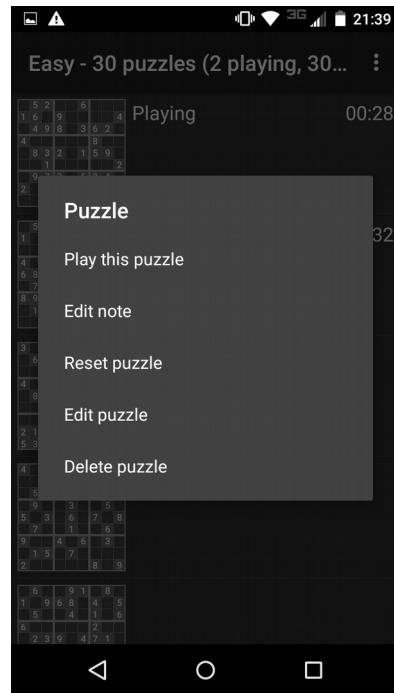
From Google Play

The below screenshots show OpenSudoku running on my phone as downloaded from Google Play.

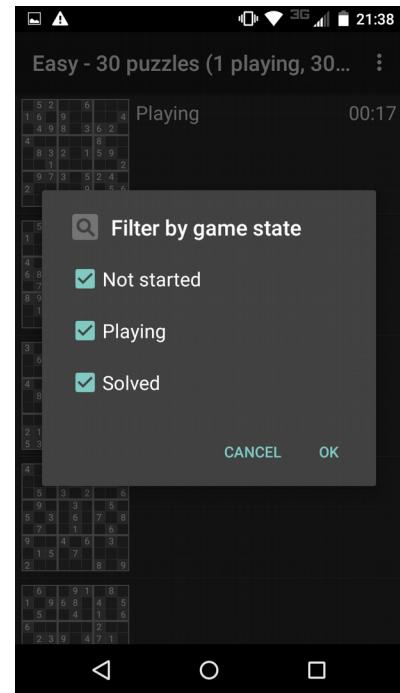
Opening Screen



Puzzle Selection Options



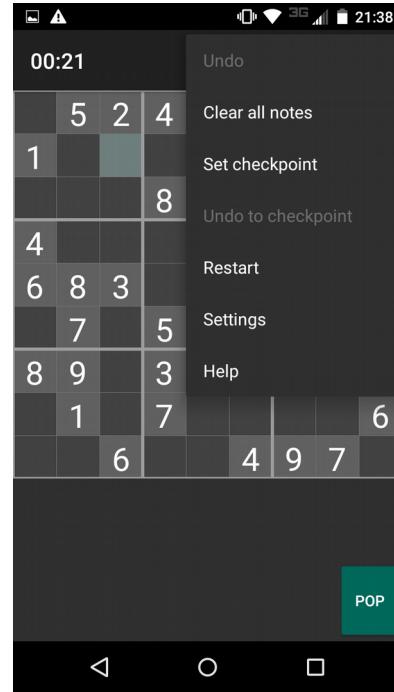
Puzzle Filter Criteria



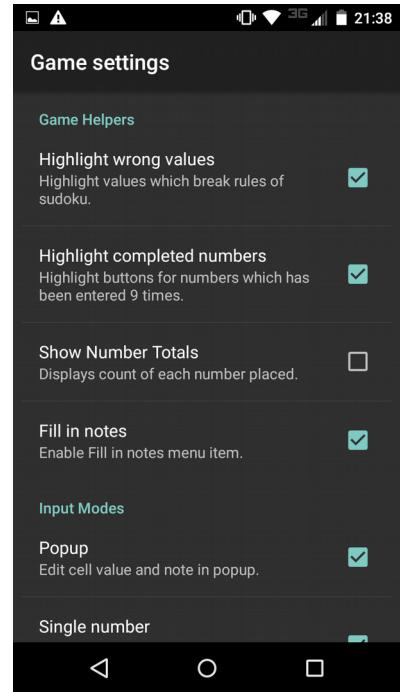
Playing a Puzzle



Some Puzzle Options



Settings



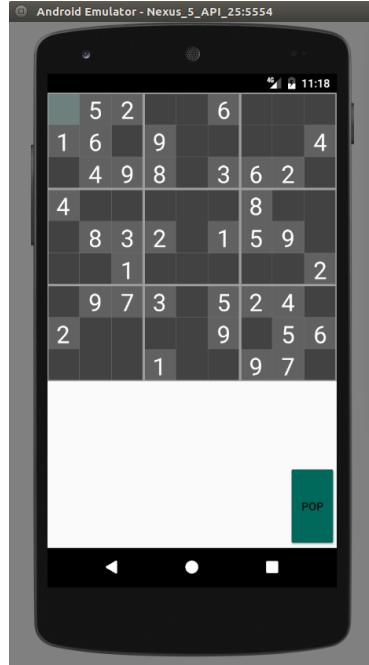
Screenshots After Pruning

The code for playing the sudoku game is unchanged. However, the opening screen was changed and the ability to choose, save, import, and export puzzles was removed.

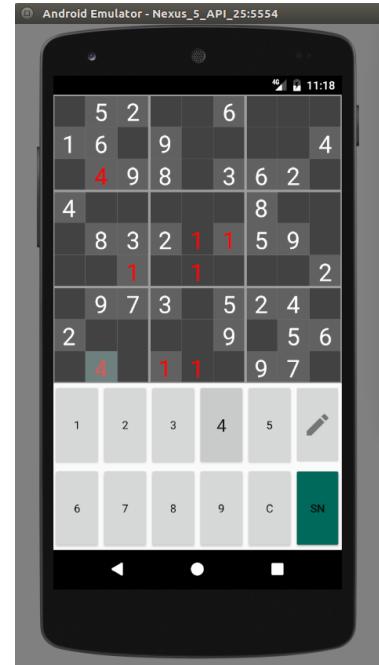
New Opening Screen



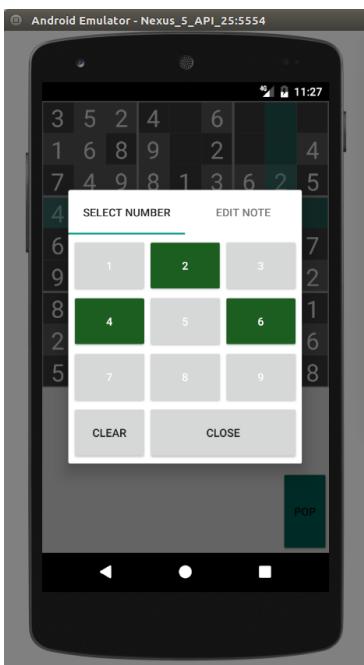
Initial Position



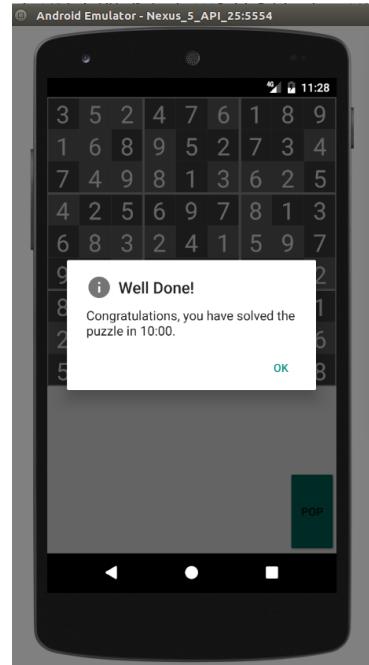
Entering Some Numbers



Almost Finished



Victory



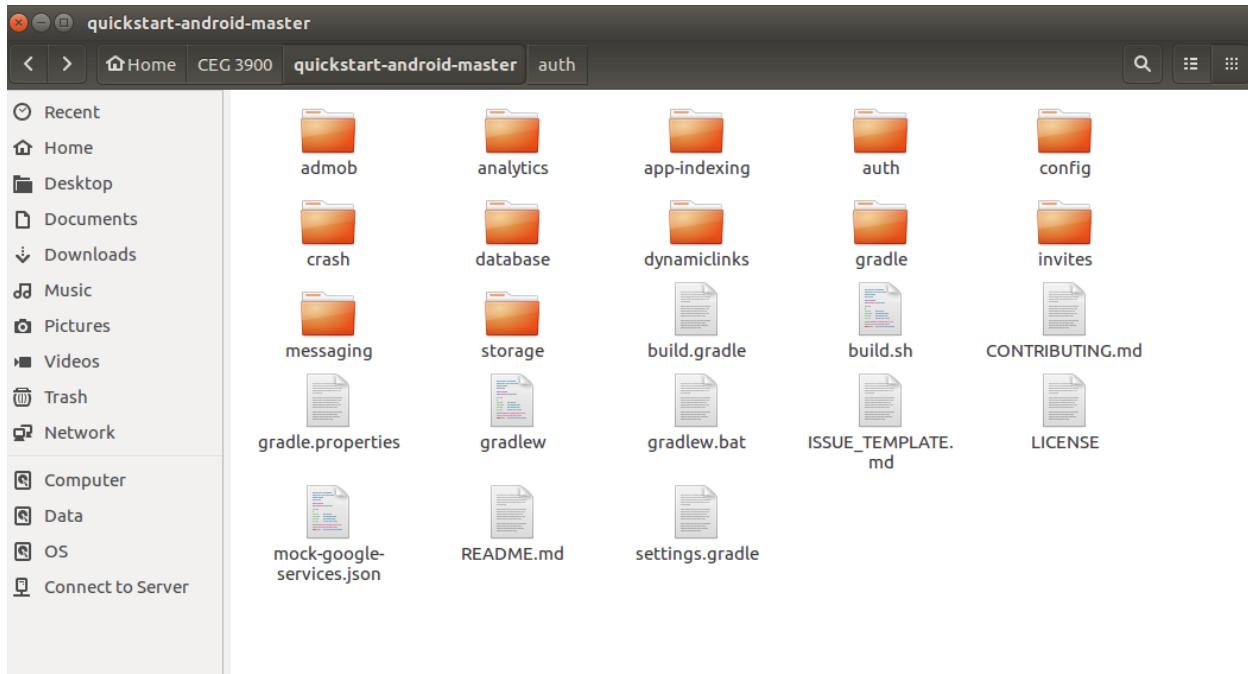
Firebase Getting Started

Downloading the github project

The screenshot shows the GitHub repository page for 'firebase / quickstart-android'. At the top, there are navigation links for Personal, Open source, Business, Explore, Pricing, Blog, Support, and a search bar. There are also 'Sign in' and 'Sign up' buttons. Below the header, the repository name 'firebase / quickstart-android' is displayed, along with metrics: 210 issues, 1,678 stars, and 1,596 forks. A 'Watch' button is also present. The main content area shows the repository's activity: 333 commits, 3 branches, 0 releases, 21 contributors, and an Apache-2.0 license. A 'Clone or download' button is visible on the right. The commit history lists several pull requests, including 'Add email verification' by samtstern, which has been merged. Other commits mention bumping the Firebase SDK version to 10.0.1 and updating README.md.

Structure of firebase quick start

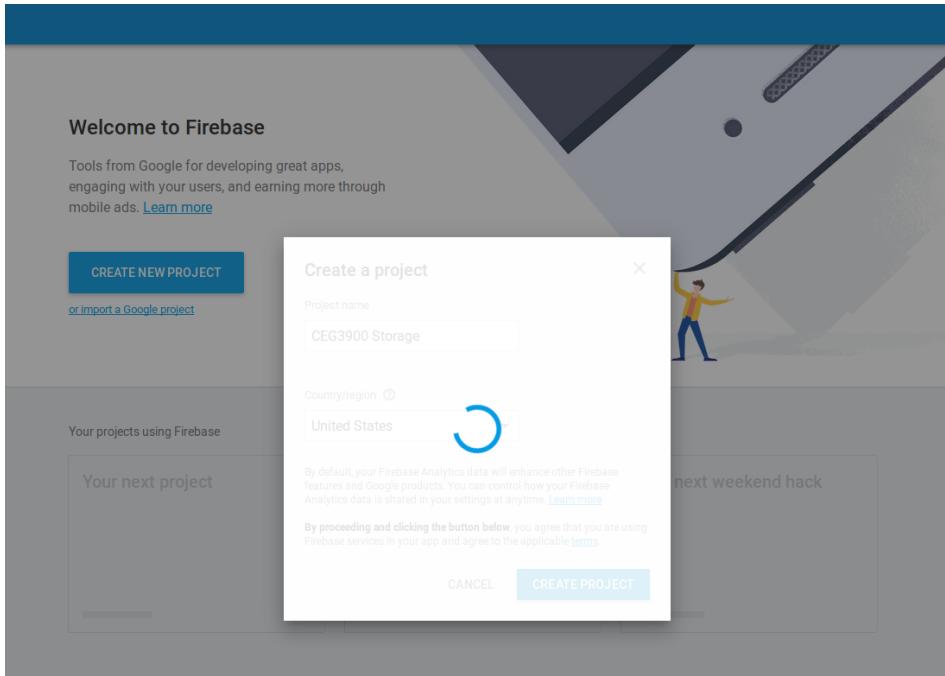
The github contained several android studio projects. For this assignment, I will be building and running the auth and storage projects.



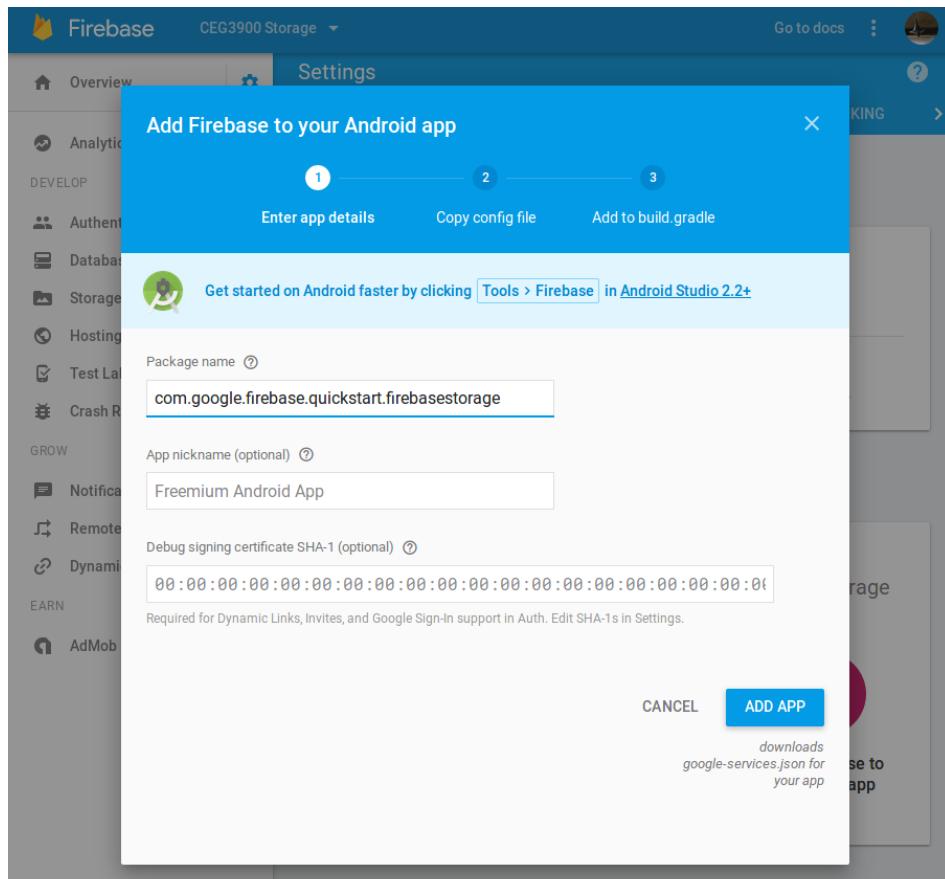
3.3 Firebase Storage

Setting up the firebase account

I created a firebase project for the storage app



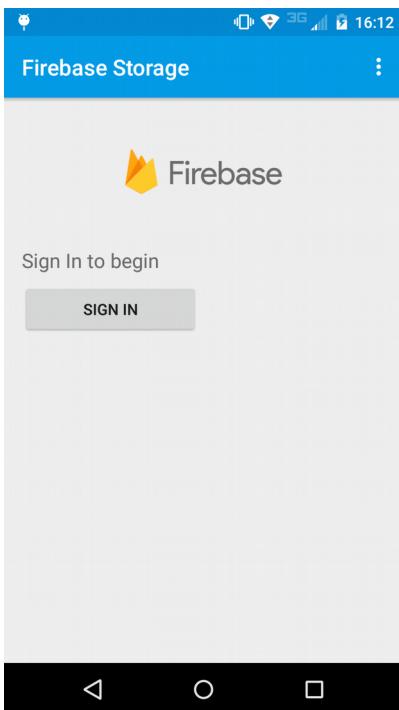
Under project settings, I got the link to and downloaded the google services json file



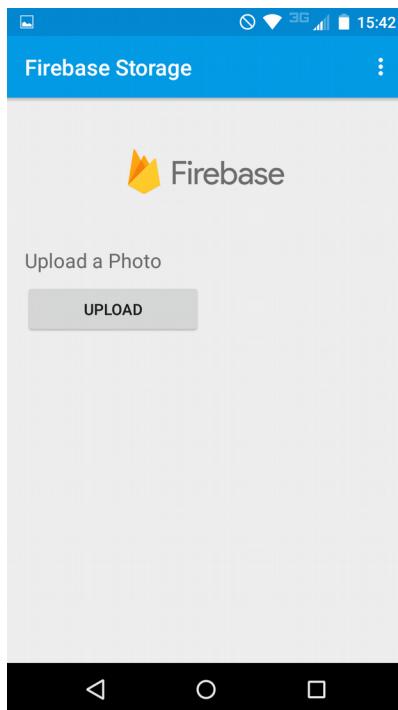
Running firebase storage

To compile the apk, I updated gradle and sdk version numbers to match the ones I have installed. I was then able to run the apk on my phone (it did not work on my emulator because of its google play version). Below are screenshots showing screens of the running app.

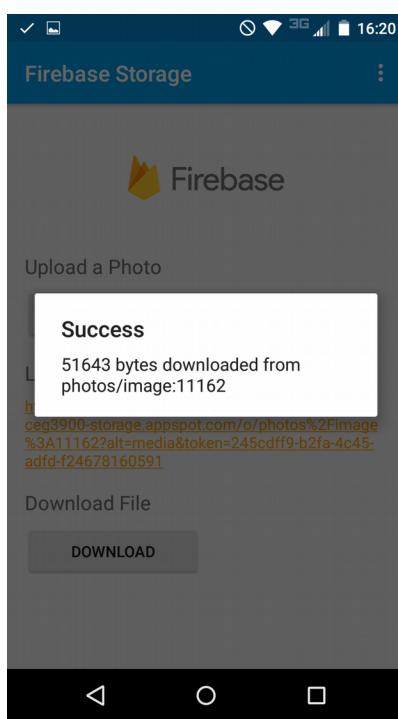
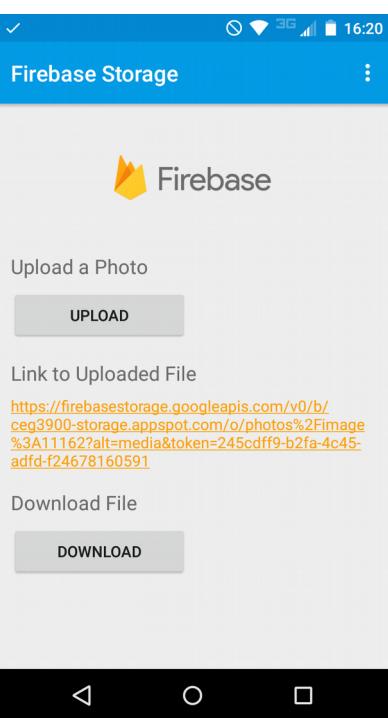
Sign in Screen



Upload Photo Screen



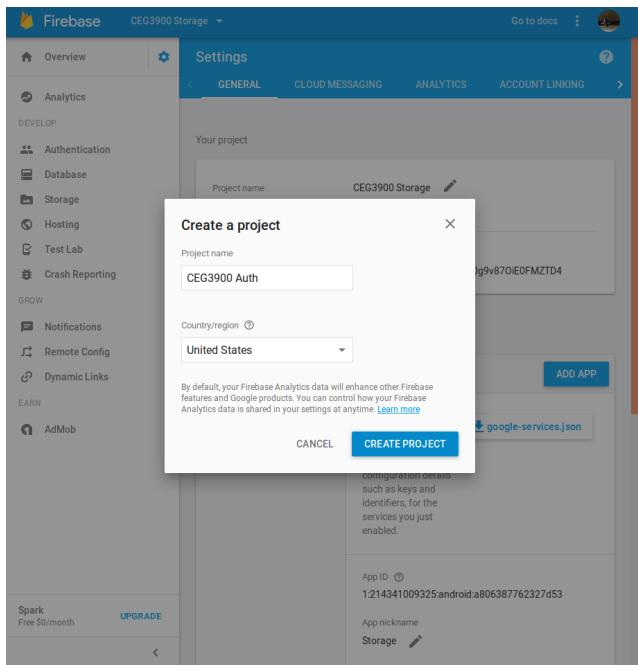
Choosing Photo to Upload



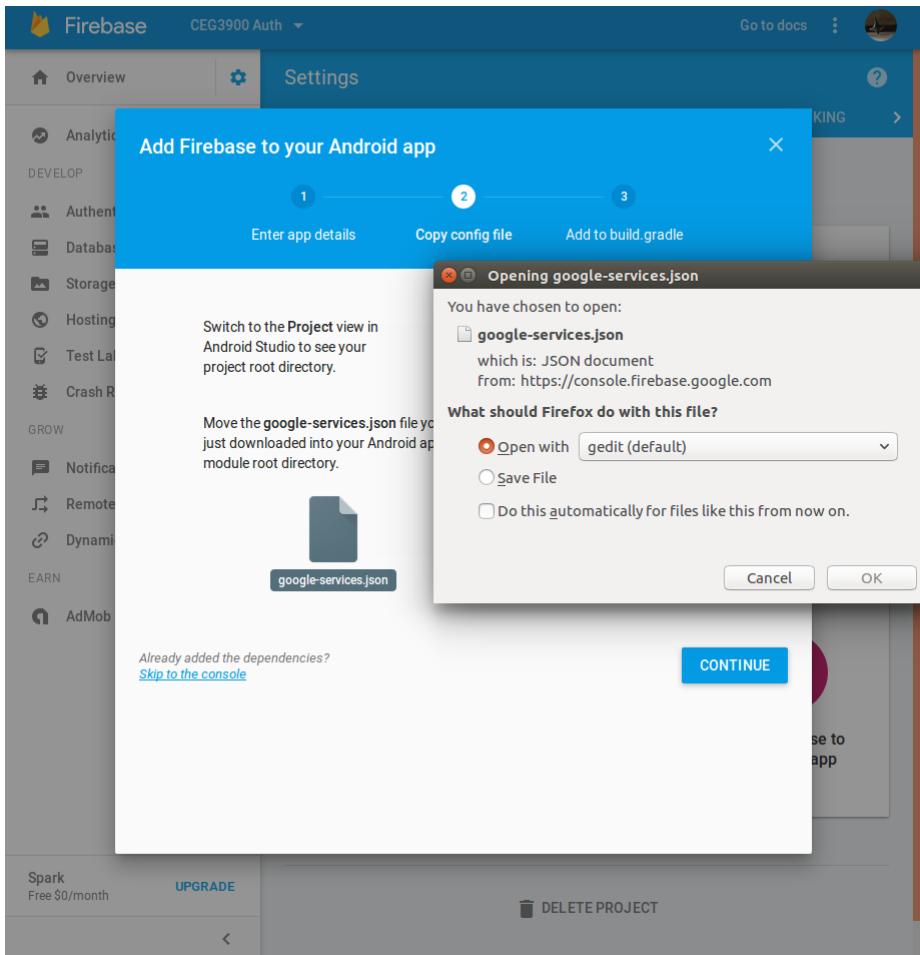
Download Photo Screen
Downloaded Photo from Firebase

3.4 Firebase Auth

Creating the firebase project

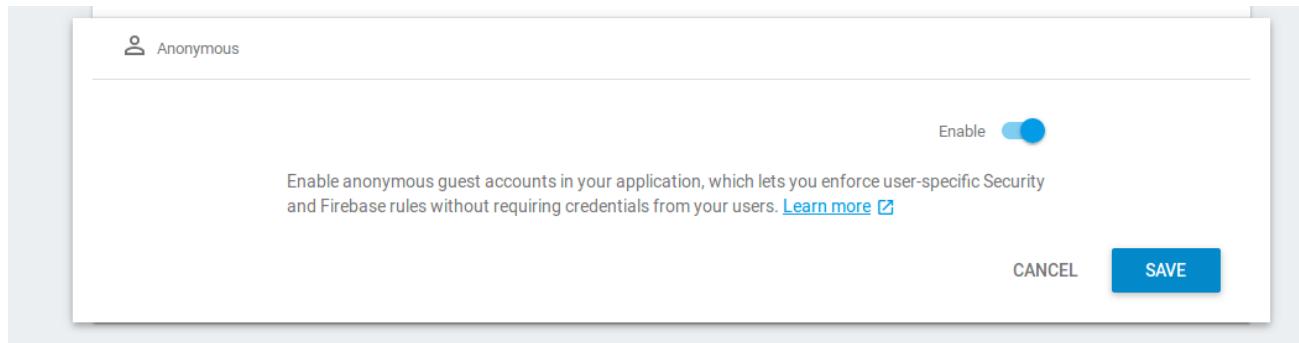


Downloading the google-services.json file



Choosing authentication options in firebase under Console -> Project ->Authentication. For this project, I tried both Google and Anonymous authentication.

Screenshot of enabling an authentication method in firebase

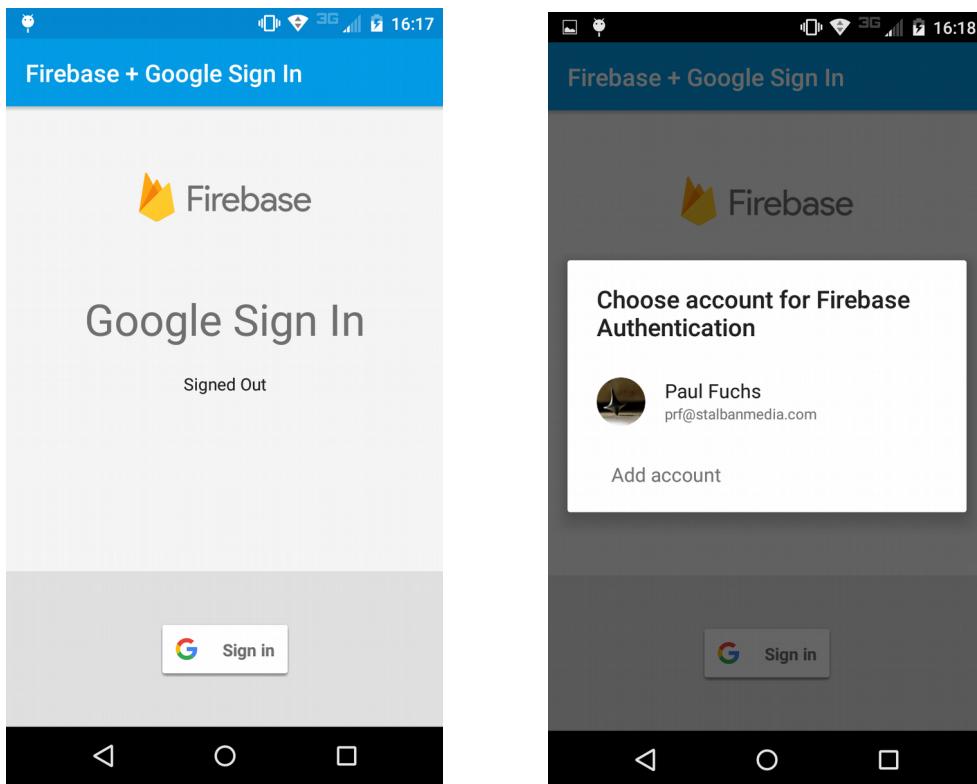


Below shows the final authentication setup for the project

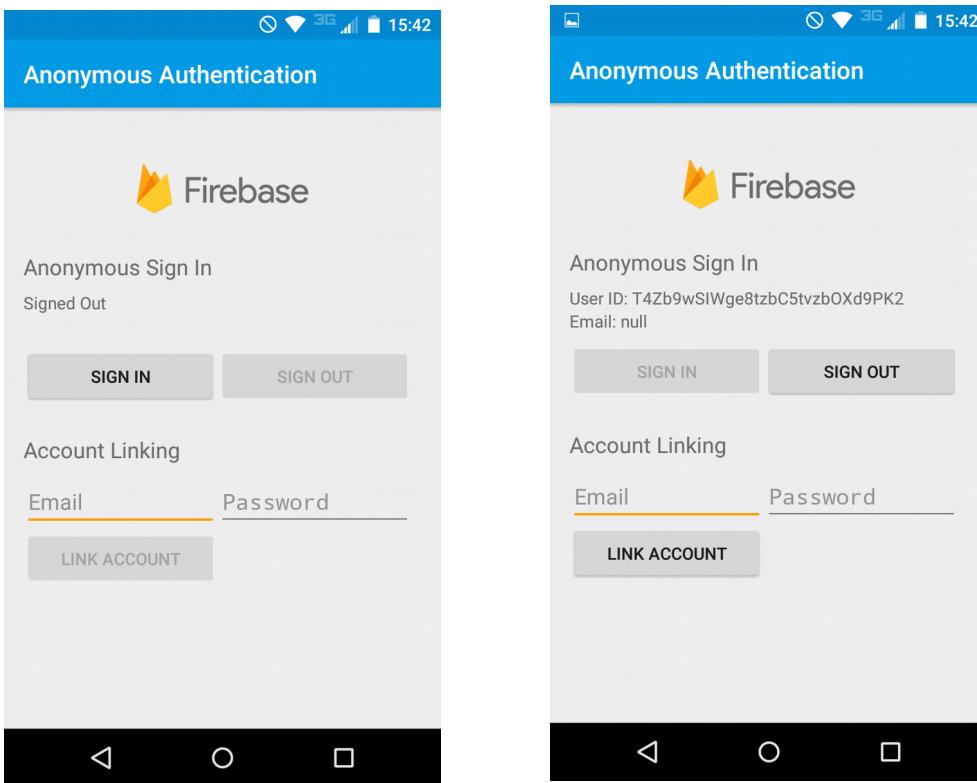
Sign-in providers	
Provider	Status
Email/Password	Disabled
Google	Enabled
Facebook	Disabled
Twitter	Disabled
GitHub	Disabled
Anonymous	Enabled

Running the apk

Google sign-in. This one did not seem to work as the app continued to display “Signed Out” after I selected my google account.



Below are screenshots showing anonymous sign-in. After sign-in a User ID was displayed.



3.5 GitHub

The files for this project will be located in the repository <https://github.com/helmmhammerhand/ceg-3900-spring-2017>. They will be uploaded sometime on Saturday 11 February after the project's due date. The below screenshot shows the (empty) GitHub repository I created for this class.

The screenshot shows a GitHub repository page. At the top, there is a header with a search bar, pull requests, issues, and gist links. Below the header, the repository name 'helmmhammerhand / ceg-3900-spring-2017' is displayed, along with watch, star, and fork counts (all 0). A navigation bar below the repository name includes links for Code, Issues (0), Pull requests (0), Projects (0), Wiki, Pulse, Graphs, and Settings. The main content area has a section titled 'Quick setup — if you've done this kind of thing before' with instructions for cloning the repository via HTTPS or SSH, and a note about including README, LICENSE, and .gitignore files. It also provides command-line setup instructions and a ProTip for adding GitHub as a remote. Other sections include '...or create a new repository on the command line' with a code snippet, '...or push an existing repository from the command line' with another code snippet, and '...or import code from another repository' with a 'Import code' button.

Quick setup — if you've done this kind of thing before

or [HTTPS](https://github.com/helmmhammerhand/ceg-3900-spring-2017.git) [SSH](https://github.com/helmmhammerhand/ceg-3900-spring-2017.git) <https://github.com/helmmhammerhand/ceg-3900-spring-2017.git>

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# ceg-3900-spring-2017" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/helmmhammerhand/ceg-3900-spring-2017.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/helmmhammerhand/ceg-3900-spring-2017.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

ProTip! Use the URL for this page when adding GitHub as a remote.